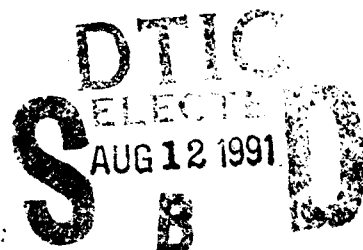


2

AD-A239 382



**Design and Analysis of
Fault-Tolerant Distributed Real-Time Computer Systems**

Final Report
Contract No. N00014-86-K-0392

July 23, 1991

Project Director : K. H. Kim
Contractor : University of South Florida
Division of Sponsored Research
Tampa, Florida 33620
Sponsor : Office of Naval Research
800 North Quincy Street
Arlington, Virginia 22217-5000
Project Monitor : Dr. Gary M. Koob, Code 1133



Computer Science
Technical Report Series

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

91-07402

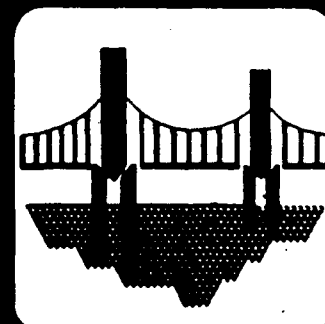
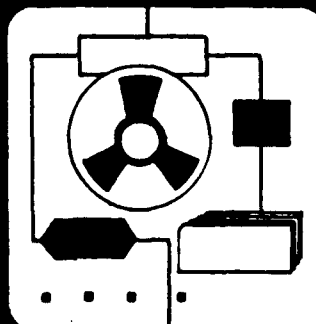
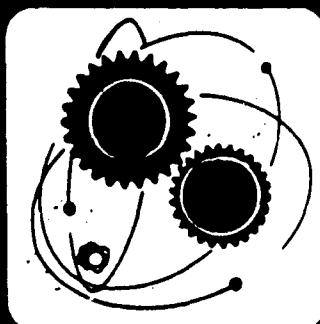
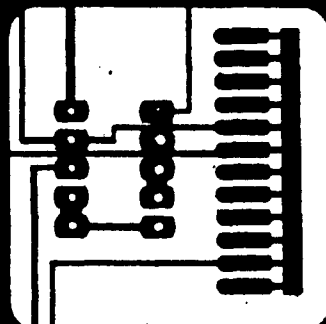


91

College of Engineering

Department of Computer Science

University of South Florida
Tampa, Florida 33620



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION / AVAILABILITY OF REPORT Unlimited	
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE				
4. PERFORMING ORGANIZATION REPORT NUMBER(S)			5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION University of South Florida	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Office of Naval Research Resident Representative		
6c. ADDRESS (City, State, and ZIP Code) Division of Sponsored Research Tampa, Florida 33620		7b. ADDRESS (City, State, and ZIP Code) Georgia Institute of Technology 206 O'Keefe Building Atlanta, Georgia 30332-0490		
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Office of Naval Research	8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0392		
8c. ADDRESS (City, State, and ZIP Code) Code 1133 800 North Quincy Street Arlington, VA. 22217-5000		10. SOURCE OF FUNDING NUMBERS		
		PROGRAM ELEMENT NO	PROJECT NO	TASK NO
		WORK UNIT ACCESSION NO		
11. TITLE (Include Security Classification) Design and Analysis of Fault-Tolerant Distributed Real-Time Computer Systems				
12. PERSONAL AUTHOR(S) K. H. Kim				
13a. TYPE OF REPORT Final Technical	13b. TIME COVERED FROM 86/06/01 to 87/06/30	14. DATE OF REPORT (Year, Month, Day) 1991, July, 23	15. PAGE COUNT 14	
16. SUPPLEMENTARY NOTATION				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP	real time computer, fault tolerance, distributed computing, recovery, temporal behavior	
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
<p>The objective of this project was to contribute to the establishment of the scientific foundation for designing fault-tolerant distributed computer systems. The results obtained from the Phase I of this research project are summarized here. Main results are as follows.</p> <p>(1) A preliminary approach to specification of timing constraints during distributed computer system design and to validation of the temporal behavior of the system,</p> <p>(2) A scheme for coordinated execution of independently designed recoverable distributed processes, and</p> <p>(3) Establishment of initial versions of three of the real-time computer network testbeds and a testbed-based evaluation of the distributed recovery block (DRB) scheme.</p> <p>The Phase II of this project was conducted at the University of California, Irvine under Contract No. N00014-87-K-0231 and concluded on Dec. 31, 1989.</p>				
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL K. H. Kim			22b. TELEPHONE (Include Area Code) (714) 856-5542	22c. OFFICE SYMBOL

**Design and Analysis of
Fault-Tolerant Distributed Real-Time Computer Systems**

Final Report
Contract No. N00014-86-K-0392

July 23, 1991

Project Director : K. H. Kim

Contractor : University of South Florida
Division of Sponsored Research
Tampa, Florida 33620

Sponsor : Office of Naval Research
800 North Quincy Street
Arlington, Virginia 22217-5000

Project Monitor : Dr. Gary M. Koob, Code 1133

"Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the US Government.

Table of Contents

Section

1. Introduction
2. Research Directions
3. Results of the Phase-I Research Conducted at USF
4. Conclusion and an Abstract of the Phase-II Research Conducted at UCI

Appendix A.

The Current Status of the Real-Time Distributed Computing Testbeds Established



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

1. Introduction

This report summarizes the main results of our research carried out at the University of South Florida (USF) under Contract No. N000014-86-K-0392-P00001 during the period of June 1, 1986 - June 30, 1987. The project was motivated by the recognition that designing real-time distributed computer systems (DCS's) had been largely an artistic activity and a scientific foundation for reliable and systematic design existed only in a weak and incoherent state. Such foundation has become an important research issue in computer science and engineering due to the continuous increase in demands for ultra-reliable computer systems capable of supporting critical real-time applications. While the long term objective of this project was to contribute to the establishment of the scientific foundation for designing fault-tolerant DCS's with response time guarantee, more specific goals of the project were the following.

- (1) Establish a real-time distributed computation model yielding simple techniques for response time guarantee,
- (2) Develop DCS architectures possessing effective fault-tolerance capability, expandability, and high predictability of worst-case performance,
- (3) Develop design environments supporting specification and validation of real-time behavior.

The research reported here constituted the first phase of a two-phase project. The follow-on phase, Phase II, was conducted at the University of California, Irvine (UCI) under Contract No. N00014-87-K-0231 and concluded on Dec. 31, 1989. An abstract of the Phase-II research results is included in the concluding section.

2. Research Directions

At the early stage of this project, some design philosophies and study strategies which might distinguish this project from others were adopted. They can be summarized as follows.

- (1) Make the processing deadlines explicitly treated attributes of both atomic and compound computation units.
- (2) Distinguish formally between the real-time database and the archival database.
- (3) Pursue deterministic time behavior in designing communication protocols, operating system (OS) structures, and application software.
- (4) Build the distributed clock synchronization logic into a VLSI component in a network interface unit to achieve the microsecond level synchronization.
- (5) Explore the fault tolerance (FT) schemes that can handle in a uniform manner both hardware faults and software faults, the latter including OS faults and application software faults.
- (6) Identify the generic forward recovery techniques applicable to hard-real-time applications as clearly distinguished from others.
- (7) Reflect unique characteristics of tightly coupled networks (TCN's) (e.g., a radar-tracking parallel computer system located at a single ground site) and loosely coupled networks (LCN's) (e.g., local area networks (LAN's) in factories or wide area networks in defense applications) in developing fault tolerance schemes and OS structures.
- (8) Develop first the real-time FT schemes that can enhance the robustness of a computing station (a processing node executing a single application process) and then the supplementary schemes for making a group of cooperating computing stations fault-tolerant.
- (9) Validate the formulated architectures, OS structures, and FT schemes not only via analyses and logical reasoning/proofs but also via experimental incorporation into real-time computer network testbeds built on TCN hardware, LCN hardware, and functional real-time application models.

3. Results of the Phase-I Research Conducted at USF

Main results obtained during the reporting period (Phase I) are as follows.

3.1 A preliminary approach to specification of timing constraints during distributed computing system design and their validation

It was the premise of this research that the designs of real-time computer systems needed in critical applications must be rigorously verified for their capabilities for meeting the specified deadlines. Such designing with response time guarantee, called safe design here, is dependent upon the system configuration and the scheduling strategy used among other factors.

Specification of timing constraints is the very first step in safe design of real-time systems. One of the timing specification approaches studied may be called the time-tagged block approach. The basic idea is to specify timing constraints in association with execution blocks. During the Phase I the notion of completeness of a set of timing specification primitives needed to support the time-tagged block approach, was formalized and then a complete and practical set of primitives was established [Yan86].

Validation of time specifications is basically to check the feasibility of meeting the specifications at run-time. A preliminary version of an overall methodology for such validation for the case of distributed computer systems was formulated [Yan86]. The methodology uses various analytic verification techniques in its several constituent steps. The techniques are generally of two types: one that is machine-independent, and the other that is machine-dependent. The machine-independent techniques detect the inconsistency in the specifications and the impossibility of meeting the specifications, when the undesirable properties persist regardless of machine configurations and operating strategies. The other techniques reflect the machine characteristics in determining the execution feasibility.

3.2 A scheme for coordinated execution of independently designed recoverable distributed processes

A scheme for facilitating efficient backward recovery in loosely coupled networks (LCN's) was developed [You88]. The scheme, called the PTC/LCN (programmer-transparent coordination / LCN) scheme, is meant to be a fully general approach to facilitating efficient backward recovery in LCN environments where the autonomy of each process is highly desired. It shares the same basic design philosophy with the original PTC scheme proposed earlier in [Kim78], but was formulated to fit LCN systems unlike the original PTC scheme better suited for centralized

systems. The scheme allows independent and uncoordinated design of error detection and recovery capabilities of distributed processes. It makes provision for properly coordinating such distributed processes at run-time for cooperative recovery without incurring a cyclic chain of rollback propagations called a domino effect. The operational rules of the scheme were devised such that a minimal number of recovery-points (RP's) were used for maintaining the capability for recovery with minimum-distance rollbacks. These capabilities were formally proved.

The system design philosophy underlying the scheme is such that each process must be solely responsible for detecting the errors that it originates. An approach to making judicious exceptions, i.e., utilizing the cooperative error detection capabilities of processes without incurring a domino effect, was devised in order to further enhance the system robustness.

3.3 Testbed establishment and an evaluation of the DRB scheme

For rigorous validation of newly formulated design techniques and system structures, we adopted the approach of testbed-based validation. The availability of low-cost building-blocks such as microcomputers and interconnection devices, had made the construction of cost-effective DCS testbeds not much more expensive than constructing pure software simulators running on centralized computer systems. Testbeds are capable of representing the operating environment and input scenario more accurately than software simulators.

Initial versions of three major real-time distributed computing testbeds were established, each including a distributed real-time control program and a simulator of an application environment with sensor devices and actuators. The three testbeds deal with the three different types of real-time object tracking applications: (1) tracking with a ground-based radar, (2) tracking with a sensor boarded on a high speed moving vehicle, and (3) cooperative tracking by sensors distributed over multiple satellites. The first testbed was built around an in-house developed tightly coupled microcomputer network called the Macro Dataflow Network (MDN), the second testbed around another tightly coupled microcomputer network built by Unisys Corp., called the Crossbar Multi-microcomputer System (CMS), and the third testbed around a local area network manufactured by Cromemco Inc.

All the real-time distributed operating systems used in the three testbeds were developed in house. New techniques and tools were to be evaluated by integrating them into the testbed facilities and applying them to the experimental development of a practical network application system.

One of the design techniques evaluated by use of the MDN testbed is the distributed recovery block (DRB) scheme initially formulated in [Kim84]. The DRB scheme is based on a combination of both distributed concurrent processing and recovery block structuring concepts to achieve fast forward error recovery and to treat both hardware and software faults in a uniform manner with minimal execution overhead. It is an active redundancy scheme where multiple processors concurrently execute multiple versions of a software component and then the same acceptance test. The acceptance test in each processor, together with a watch-dog timer, checks reasonableness of the computational results of the version executed as well as the timeliness of the execution. The scheme was incorporated into the MDN testbed and subsequent measurement and evaluation demonstrated the fast recovery capability of the scheme and the soundness of the implementation strategies adopted [Yoo88].

The testbed facilities were transferred to UCI in early 1987 and have since been upgraded in major ways. The current status of the testbeds established is described in Appendix A.

4. Conclusion and an Abstract of the Results of the Phase-II Research Conducted at UCI

The results summarized in this report represent some advances in the state of the art in the design of fault-tolerant real-time DCS's. More importantly, they established directions for specific research which were pursued more extensively during Phase II, thereby resulting in substantial addition to the knowledge base related to fault-tolerant real-time distributed computing. Main results obtained during the Phase II are as follows.

- (1) Identification of critical research issues and some promising research directions in real-time fault-tolerant distributed computing,
- (2) A skeleton of the foundation for realizing system-level fault tolerance, which includes among others the DRB (distributed recovery block) scheme, the DCONV (distributed conversation) scheme, the PTC (programmer-transparent coordination) scheme, a TB (temporary blackout) handling scheme, and the complementary relationship among the schemes; These schemes enable the computer system to detect and recover from both hardware and software faults without missing the deadlines for processing important data and delivering outputs to the controlled object/environment,
- (3) A preliminary structure of a model of real-time distributed computation,
- (4) A theoretical investigation into the efficiency and diagnostic power of basic processor-level diagnosis approaches in diagnosing hypercubes conducted,
- (5) An enhancement of three of the real-time computer network testbeds established in the UCI DREAM (Distributed Real-time Ever Available Microcomputing) Laboratory made.

Details of these results are available from the publications listed in Section 5.2.

5. References

5.1 Publications from the Phase-I research conducted at USF

[Yan86] Yang, S.M., 'Timing Specification and Verification for Fault-Tolerant Distributed Computer Systems', Ph.D. Dissertation, Dept. of Computer Sci. & Engr., Univ. of South Florida, Dec. 1986.

[Yoo88] Yoon, J.C., 'An Approach to Design of Fault-Tolerant Real-Time Tightly Coupled Networks and Its Experimental Validation', Ph.D. Dissertation, Dept. of Computer Sci. & Engr., Univ. of South Florida, May 1988.

[You88] You, J.H., 'Techniques for Efficient Implementation of Fault-Tolerant Loosely Coupled Networks Based on Decentralized Computation Recovery', Ph.D. Dissertation, Dept. of Computer Sci. & Engr., Univ. of South Florida, May 1988.

5.2 Publications from the Phase-II research conducted at UCI

[Kav91] Kavianpour, A. and Kim, K.H., "Diagnosabilities of Hypercubes under the Pessimistic One-Step Diagnosis Strategy", IEEE Transactions on Computers, Vol.40, No.2, Feb. 1991, pp.232-237.

[Kim88a] Kim, K.H. and Yoon, J.C., "Approaches to Implementation of a Repairable Distributed Recovery Block Scheme", Proc. IEEE Computer Society's 18th Int'l Symp. on Fault-Tolerant Computing, June 1988, pp.50-55.

[Kim88b] Kim, K.H., "Programmer-Transparent Coordination of Recovering Parallel Processes: Philosophy and Rules for Efficient Implementation" IEEE Trans. on Software Engineering, Vol.14, No.6, June 1988, pp.810-821.

[Kim88c] Kim, K.H., "Designing Fault Tolerance Capabilities into Real-Time Distributed Computer Systems", Proc. IEEE Computer Society's Workshop on the Future Trends of Distributed Computing Systems in the 1990's, Hong Kong, Sep. 1988, pp.318-328.

[Kim88d] Kim, K.H. and Yang, S.M., "An Analysis of the Performance Impacts of Lookahead Execution in the Conversation Scheme", Proc. IEEE Computer Society's 7th Symp. on Reliable Distributed Systems, Columbus, Oct. 1988, pp.71-81.

[Kim88e] Kim, K.H., "Issues in Design of Temporary Blackout Handling Capabilities into Tightly Coupled Computer Networks", Proc. 2nd Int'l Software for Strategic Systems Conf., Huntsville, AL, Oct. 1988, pp.92-99.

[Kim88f] Kim, K.H., "Design of Fault-Tolerant Distributed Computer Systems with Response Time Guarantee", Proc. ONR Foundations of Real-Time Computing Research Initiative Kickoff Workshop, Falls Church, VA, Nov. 1988, pp.101-106.

[Kim89a] Kim, K.H. and Welch, H.O., "Distributed Execution of Recovery Blocks: An Approach to Uniform Treatment of Hardware and Software Faults in Real-Time Applications", IEEE Trans. on Computers, May 1989, pp.626-636.

[Kim89b] Kim, K.H., "An Approach to Experimental Evaluation of Fault-Tolerant Distributed Computing Schemes", IEEE Transactions on Software Engineering, June 1989, pp.715-725.

[Kim89c] Kim, K.H. and Yang, S.M., "Performance Impacts of Look-Ahead Execution in the Conversation Scheme", IEEE Transactions on Computers, August 1989, pp.1188-1202.

[Kim89d] Kim, K.H., "Approaches to System-Level Fault Tolerance in Distributed Real-Time Computer Systems", Proc. 4th Int'l Conf. on Fault-Tolerant Computing Systems, Baden-Baden, W. Germany, Sept. 1989, pp.268-281 (published in the Lecture Notes series by Springer-Verlag) (Invited paper).

[Kop88] Kopetz, H. and Kim, K.H., "Consistency Constraints in Distributed Real Time Systems", in M.G. Rodd and T.L. d'Epinay eds., 'Distributed Computer Control Systems 1988', Pergamon Press, 1988 (Proc. 8th IFAC Workshop on Distributed Computer Control Systems, Vitznau, Swiss, Sep. 1988), pp.29-34.

[Min91] Min, B.J., "Efficient Implementation of the Distributed Recovery Block (DRB) Scheme in Highly Parallel Computer Systems", Ph.D. Dissertation, Dept. of Electrical & Computer Engineering, Univ. of Calif., Irvine, Aug. 1991.

[Yan89] Yang, S.M. and Kim, K.H., "Implementation of the Conversation Scheme in Loosely Coupled Distributed Computer Systems", Proc. IEEE Computer Society's 9th Int'l Conf. on Distributed Computing Systems, June 1989, pp.570-578.

[Yan91] Yang, S.M. and Kim, K.H., "Implementation of the Conversation Scheme in Message-Based Distributed Computer Systems", Tech. Rept. UCI-ECE-90-12, Dept. of Electrical & Computer Engineering, UCI, October, 1990. To appear in IEEE Transactions on Parallel and Distributed Systems.

5.3 General references

[Kim78] Kim, K.H., "An Approach to Programmer-Transparent Coordination of Recovering Parallel Processes and Its Efficient Implementation Rules", Proc. 1978 Int'l Conf. on Parallel Processing, August 1978, pp.58-68.

[Kim84] Kim, K.H., "Distributed Execution of Recovery Blocks: an Approach to Uniform Treatment of Hardware and Software Faults", Proc. 4th Int'l Conf. on Distributed Computing System, May 1984, pp.526-532.

[Kim86] Kim, K.H., You, J.H., and Abouelnaga, A., "A Scheme for Coordinated Execution of Independently Designed Recoverable Distributed Processes", Proc. IEEE Computer Society's 16th Int'l Symp. on Fault-Tolerant Computing, July 1986, pp.130-135.

[Ran75] Randell, B., "System Structure for Software Fault Tolerance", IEEE Transactions on Software Engineering, June 1975, pp.220-232.

Appendix A
The Current Status of
the Real-Time Distributed Computing Testbeds Established

As a part of the experimental work, the PI has established a laboratory named the DREAM Laboratory. The laboratory was started in the PI's former institution, Univ. of South Florida, in 1980 and was moved to his current institution, Univ. of California, Irvine (UCI), in January 1987.

The DREAM Lab consists of the Loosely Coupled Network (LCN) Section and the Tightly Coupled Network (TCN) Section. There are two major testbeds established in each section. The equipment in the LCN Section includes a Cromemco LAN connecting four MC68000-based microcomputers and a LAN of four 80386-based PC's made by AST Research Inc. and connected by Ethernet. The Cromemco LAN is several years old and the PC LAN was established in 1990. These form the first testbed in the LCN Section. The operating system and real-time application software running on the Cromemco LAN has been almost fully transported to the PC LAN with some restructuring of the operating system. The second testbed was added in the Fall of 1989. A 3-node ADS (Autonomous Decentralized System) made by Hitachi was established. The ADS has a unique communication architecture called the data field that enables easy expansion and reconfiguration. Each node is based on M68020 and has a unique network interface. One node runs a UNIX-ACP combination and two other nodes run the ACP (Atom Control Program) which is a Hitachi's proprietary real-time operating system.

The TCN Section includes two major networks: (1) One called the Macro-Dataflow Network (MDN) is a homemade network of six single-board Z8001-based microcomputers connected through up to 12 two-port buffer memory modules and (2) the other called the Crossbar Multi-microcomputer System (CMS) is a network of seven single-board microcomputers and five multi-port shared memory modules connected through a crossbar connection subsystem and manufactured by the Unisys Corporation in Huntsville, Alabama.

Real-time distributed operating systems and distributed application programs have been developed to run on three computer networks in the DREAM Lab., i.e., PC LAN, MDN, and CMS, and a real-time application program was added to run under the manufacturer's operating system of the ADS. The application programs included in the two TCN testbeds and the PC/Cromemco testbed are the distributed real-time control programs combined with simulators of applications environments with sensor devices and actuators. The three testbeds deal with the three different types of real-time object tracking applications: (1) tracking with a ground-based radar (the MDN testbed), (2) tracking with a sensor boarded on a high-speed moving vehicle (the CMS testbed), and (3) cooperative tracking by sensors distributed over multiple satellites (the PC/Cromemco testbed, also called the Defense Satellite Network testbed). The PC/Cromemco

testbed thus deals with a WAN application although the hardware base used is a LAN. It is a kind of dual-purpose LAN/WAN testbed.

The three testbeds dealing with object tracking applications were used in conducting the fault tolerance experiments discussed in the main report. Figure 1 provides a brief summary of the current status of the three testbeds.

Quite a few software and hardware tools for prototyping of real-time computer networks have been established in the UCI DREAM Laboratory. They include operating system components, communication primitives, and high level languages such as C, Extended Concurrent Pascal, Modula-2, and Unisys PDL. There are also tools for measurement of message delays. An approach formulated for rapid prototyping of software for real-time computer networks is a two-step approach in which the first inefficient version is obtained with the aid of an abstract high level language such as Extended Concurrent Pascal or ADA, and then using the first version as a blueprint, the final version is written in an efficient language such as C. This approach has been partially tested in the DREAM Laboratory with good results. The main approach established in the DREAM Laboratory for network performance measurement is to install "observation points" within a network such that when a message passes through an observation point, a time-stamped record is made by an observing machine. By comparing the time-stamped records made at different observation points, the time taken for a message to travel between observation points can be obtained.

PC-based facilities for graphic display of the run-time status of both the distributed application software and the hardware configuration have been established as integral components of both testbeds (CMS and MDN) in the TCN Section.

Real-Time Defense Computer Network Testbeds Established

- **Terminal defense controller (TDC) testbed**
 - simulates ground-based radar tracking activities
 - built on the six-node tightly coupled microcomputer network called the MDN (Macro-Dataflow Network)
 - uses an OS developed in house
 - about 10K lines of Extended Concurrent Pascal, C, and Z8001 assembly code
- **On-board intelligence (OBI) testbed**
 - simulates interceptor-borne optical sensor and data processor activities
 - built on the seven-node tightly coupled microcomputer network called the CMS (Crossbar Multi-microcomputer System)
 - uses an OS developed in house
 - about 5K lines of C and Z8001 assembly code
- **Defense satellite network (DSN) testbed**
 - simulates a squad of mid-course satellite-borne radar tracking processors
 - built initially on the four-node local area network of Cromemco Z2/68000 microcomputers
 - uses an OS developed in house
 - about 10K lines of Extended Concurrent Pascal, C, and M68000 assembly code
 - Entire software was first ported to a LAN of Intel 80386-based PC's in 1990.

Figure 1. An overview of the three real-time defense computer networks established in the UCI DREAM Lab.

Distribution List

Addressee	Copies
Scientific Officer:	1
Dr. Gary M. Koob	
ONR, Code 1133	
800 North Quincy Street	
Arlington, VA 22217	
Administrative Contracting Officer:	1
Office of Naval Research Resident Representative	
Georgia Institute of Technology	
206 O'Keefe Building	
Atlanta, Georgia 30332-0490	
Director, Naval Research Laboratory	1
Attn: Code 2627	
Washington, D.C. 20375	
Defense Technical Information Center	12
Bldg. 5, Cameron Station	
Alexandria, VA 22314	